

Appl. No. : 09/553,586  
Filed : April 20, 2000

A<sup>1</sup> that provides information or services, and a "client application" that receives the provided information and services. A client application executes on a client computer or client. A server application is executed on a server computer or server.

---

Please replace the paragraph appearing at page 12, lines 10-25 with:

---

A<sup>2</sup> In an embodiment of the invention, a control mechanism or control is implemented as an object-oriented object. The object-oriented object can be defined using the Java programming language, for example. In an embodiment of the invention, the object is a Java bean. In a design environment, the beans' appearance and behavior can be customized by a GUI or application developer. In an embodiment of the invention, a design tool is used to design a page, screen or other display. When a page design is finalized, a page object class definition is compiled and a class file is generated. Other controls defined for the page are serialized and uploaded to the server, along with any HTML that is specified at design time, to the server. A page designer can therefore generate a page control, serialized controls, and HTML in an embodiment of the invention. In one or more embodiments of the invention, the designer is executed on a client computer and the page designer (e.g., a page control, serialized controls and/or HTML) are uploaded to a server computer.

---

Please replace the text appearing at page 15, lines 2-3 with:

---

A<sup>3</sup> Figure 1 provides an example of a transfer of data between a client and server using name-value pairs (prior art).

---

Appl. No. : 09/553,586  
Filed : April 20, 2000

Please replace the text appearing at page 15, lines 4-5 with:

A<sup>4</sup>  
Figure 2 illustrates the compile and runtime environments for a processing system (prior art).

Please replace the paragraph appearing at page 16, lines 9-20 with:

A<sup>5</sup>  
In an embodiment of the invention, components of a page are modeled as control mechanisms. According to an embodiment of the invention, a control mechanism is implemented as program code such as an object-oriented object. The control mechanism is capable of generating some or all of a GUI definition (e.g., HTML statements that define a Web page). A page design environment can contain representations of control mechanisms that can be included in a page definition. For example, a palette can contain control mechanisms that may be dragged into a graphical representation of a page in the page design environment. A graphical representation of the GUI element associated with the control mechanism is displayed in the page, for example. The properties and behavior of the control mechanism may be modified in the page design environment.

Please replace the paragraph appearing at page 19, lines 3-12 with:

A<sup>6</sup>  
Figures 4A-4B illustrate a use of name spaces according to an embodiment of the invention. Referring to Figure 4A, name spaces 406, 410 and 430 are created on server 402 for controls 408A, 408B and 432. Controls 408A and 408B are, in this example, instances of the same object-oriented "Text" control object class. However, controls 408A and 408B can be instantiated from different object classes. Control 432 is an instance of an input control object class. Controls 408A, 408B and

Appl. No. : 09/553,586  
Filed : April 20, 2000

A<sup>6</sup> 432 are capable of generating definitional statements that can be used to construct GUI elements 416, 418 and 440 for display in a display area of, for example, browser 422.

---

Please replace the paragraph appearing at ~~page 20, line 20~~ to ~~page 21, line 2~~ with:

A<sup>7</sup> The name portion of the name-value pairs 426, 428 and 446 includes the label generated by controls 408A, 408B and 432, which includes the name space designations (e.g., "nS1", "nS2" and "nS3"). Name-value pairs 426, 428 and 446 are transmitted to server 402. The labels in the name-value pairs 426, 428 and 446 can be used to direct name-value pairs 426, 428 and 446 to the appropriate controls (e.g., controls 408A, 408B and 432, respectively) for processing the data.

---

Please replace the paragraph appearing at ~~page 21, lines 3-5~~ with:

A<sup>8</sup> As is discussed below, name-value pairs 426, 428 and 446 may be transmitted as events to controls 406, 410 and 432, respectively, in one or more embodiments of the invention.

---

Please replace the paragraph appearing at ~~page 21, lines 6-8~~ with:

A<sup>9</sup> To provide a better understanding of using name spaces in a graphical user interface, an overview of object-oriented programming, the Java programming language and program execution are provided below.

---

**Appl. No.** : 09/553,586  
**Filed** : April 20, 2000

Please replace the paragraph appearing at page 25, lines 6-14 with:

A<sup>10</sup>  
The classes of a Java applet are loaded on demand from the network (stored on a server), or from a local file system, when first referenced during the Java applet's execution. The virtual machine locates and loads each class file, parses the class file format, allocates memory for the class's various components, and links the class with other already loaded classes. This process makes the code in the class readily executable by the virtual machine. Native code, e.g., in the form of a dynamic linked library (DLL) 211 is loaded when a Java programming language class file containing the associated native method is instantiated within the virtual machine.

Please replace the paragraph appearing at page 33, lines 3-10 with:

A<sup>11</sup>  
In one embodiment of the invention, page control 450 may send the data to a control in the form of an event. Data change events 434, 436 and 438 are sent to controls 408A, 408B and 432, for example. As is discussed below, a portion of control 432's response may be to notify page control 450 (e.g., notification 454) that control 432 controls the GUI element used to submit the page. Page control 450 retains this information, and sends button-click event 452 to control 432 after all of the data has been processed via the data change events.

Please replace the paragraph appearing at page 36, lines 17-21 with:

A<sup>12</sup>  
Figure 5A illustrates an example of a component structure of a design of a page according to an embodiment of the invention. Page 502 includes container 504 and subpage 506. Container 504

**Appl. No.** : 09/553,586  
**Filed** : April 20, 2000

A<sup>12</sup> contains header 508 and body 510 elements. Subpage 506 includes container 512 header 514 and body 516.

---

Please replace the paragraph appearing at ~~page 36, line 22~~ to ~~page 37, line 4~~ with:

---

A<sup>13</sup> At runtime, controls are instantiated for the components of the page as illustrated in Figure 5B. Page control 522 corresponds to page 502. Container control 524 and subpage control 526 correspond to container 504 and subpage 506, respectively. Header controls 528 and 534 correspond to header 508 and 514, respectively. Body controls 530 and 536 correspond to body 510 and 516, respectively.

---

Please replace the paragraph appearing at ~~page 38, lines 3-10~~ with:

---

A<sup>14</sup> Subpage control 52 corresponds to subpage control 506, the second element in page 502. Its name space designation includes page control 522's name space designation (i.e., "s1"), followed by "\_" followed by "2" (i.e., "s1\_2"). As the first element in subpage 506, container 512's control (i.e. container control 532) is given subpage control 526's name space designation (i.e., "s1\_2"), followed by "\_" followed by "1". Header control 534 and body control 536 have name space designations of "s1\_2\_1\_1" and "s1\_2\_1\_2", respectively.

---

Please replace the paragraph appearing at ~~page 39, line 23~~ to ~~page 40, line 6~~ with:

---

A<sup>15</sup> Figure 7 provides an example of a "name-value pair" submission process according to an embodiment of the invention. At step 702, a determination is made whether all of the submitted

**Appl. No.** : 09/553,586  
**Filed** : April 20, 2000

A<sup>15</sup>  
name-value pairs have been processed. If so, processing continues at step 714 to invoke any "submit" processing (e.g., invoke a submit method of a submit control). As discussed, a control can indicate that it is to be called to perform processing once all of the data has been posted, for example. Such a control can be called at a step 714 and then end 716.

---

Please replace the paragraph appearing at ~~page 41, lines 7-15~~ with:

---

A<sup>16</sup>  
In an embodiment of the invention, a position-based name space designation assignment is used to dynamically assign name space designations. A control's position is determined, for example, by its position in a page design relative to the top left-hand corner of the page layout. The page control typically is given the first name space designation for the page (e.g., "S1"). The first control encountered receives a name space design (e.g., "1") prefixed by the page's name space designation (e.g., "s1"). Reading from left to right and top to bottom, the next controls that are encountered in the page are given the name space designation of "2" and "3", etc.

---

Please replace the Abstract appearing at ~~page 52, lines 2-22~~ with:

---

A<sup>17</sup>  
Embodiments of the invention provide a mechanism for using name spaces in graphical user interface (GUI) page definitions. A name space designation is assigned to each control mechanism that generates definitional statements for the GUI. In one embodiment, a page control is assigned a name space designation and assigns a name space designation for each of the controls that are used to generate the page. The name space designation can be used to uniquely identify a control. The